An efficient load balance using virtual machine migration hybrid optimization technique in cloud computing

Saravanan Madderi Sivalingam[1], Pavan Kumar Prathapagiri[2]
[1]Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Tamil Nadu, India
[2]Department of Computer Science and Engineering, Guru Nanak Institutions Technical Campus, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Tamil Nadu, India

Article Info

Article history:

Received Apr 12, 2024
Revised Nov 6, 2024
Accepted Nov 24, 2024

Keywords:

Gene expression programming
Genetic algorithm
Load monitoring
Physical machine
Virtual machine
Vm migration

ABSTRACT

Cloud computing is becoming increasingly important to developers and companies because to the rapid development of information technology and the wide availability of internet applications. Every information technology industry has a significant role for cloud computing. Numerous multinational technology businesses, like Google, Microsoft, and Facebook, have established data centers across the world to offer processing and storage capabilities. Customers can submit their jobs to cloud centers directly. Reducing overall power usage is the primary goal, which was overlooked in the early stages of cloud development. Using gene expression programming (GEP), symbolic regression models of virtual machines (VMs) are developed using measured VM loads and the corresponding resource parameters. In order to minimize resource use, multidimensional resource load balancing of all the physical machines within the cloud computing platform is the aim of this analysis. The VMH loads estimated and the genetic algorithm that considers the current and the future loads of VMHs and decides an optimal VM-VMH for migrating VMs and performing load-balance. Hence, an efficient load balance using virtual machine migration hybrid optimization technique (HOT) in cloud computing shows better results in terms of accuracy, energy consumption, migration cost.

Corresponding Author:

Pavan Kumar Prathapagiri
Department of Computer Science and Engineering, Guru Nanak Institutions Technical Campus
Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences
Chennai, Tamil Nadu, India
Email: pavanathraya@gmail.com

1. INTRODUCTION

Due to numerous advantages, including cost-efficiency and on-demand/pay-as-you-use services [1] that aren't dependent on time or place, cloud computing has become more and more common. A collection or set of integrated and networked hardware, software, and Internet infrastructure is referred to this general phrase. These platforms offer a very basic graphical user interface (GUI) or applications programming interface (API) to conceal the complexity and specifics of the underlying infrastructure from users and applications.

Developers and information technology (IT) professionals can now concentrate on important tasks are take care of maintenance and capacity planning duties due to cloud computing technologies. As cloud computing becomes more and more popular, several models and deployment techniques have been produced to help meet to the different requirements of various users [2]. There are differences in the levels of flexibility, control, management offered by each type of cloud service and organization technique. The platform as a service (PaaS), software as a service (SaaS), and infrastructure as a service (IaaS) are the three basic service models in cloud computing. There are four different approaches in which these three services can be implemented: private, public, community, and hybrid cloud.

Cloud computing is a resource utility that operates on the internet. The concept that "everything can be a service" is at its foundation. Through the internet, cloud computing users can access computing hardware and software resources as web services [3]. The PaaS, SaaS, and IaaS are the three application types of cloud computing models that are mentioned. One application of IaaS is virtualization, it can offer web services as that to provide computing infrastructure resources like processing power, data storage, and networking. IaaS businesses purchase, operate, and offer users web services in addition to real computing and storage infrastructure [4]. Although they show a "virtual machine" (VM) without having to buy and maintain physical hardware, users of virtualization technology can request computation or storage resources from IaaS providers. Users can use VMs to run system and application software at a much reduced hardware cost. A server with some computational storage hardware is operated by an IaaS provider, some hosting servers, also known as VMHs, provide virtualization services [5].

Depending on the virtual machine host (VMH)'s capacity, it can run one or more VMs. Some VMHs may be busy running numerous VMs while others are nearly inactive with few VMs if VMHs are not correctly handled in a server. An important problem in IaaS is managing a large number of VMHs by modifying the resources are stored by VMs to achieve improved cost-performance efficiency while providing customers with service level agreements (SLAs) [6]. One technique for load balancing VMHs is migration of VMs among VMHs. This involves moving the workload of VMs from one overloaded VMH to other VMHs in an effort to evenly distribute the workload among all VMHs. This method involves three stages of work: detection, decision-making, and action [7]. The goal of the detection phase is to find out whether a server is imbalanced. Selecting which VMs to move and which virtual machine providing (VMH) to accept them are the decisions made during the decision-making phase. It is necessary to suspend and relocate the selected VMs between VMHs, and then restarted following the migration [8]. The other VMs in the VMHs continue to run while the migration suspends and restarts the VMs that need to be transferred. As a result, the workload of VMHs is dynamic; moving VMs may result in a significant increase in the target VMHs workload. For the administration of VMHs, an efficient load balancing method is thus essential yet challenging [9].

As resource requirements increase, evaluate the VMs state of usage and transfer the underutilized machines to the target systems, a number of migration techniques are proposed. Similar to that, selecting a suitable VM for the migration process is essential, since the performance of the target application is the VM will be impacted and quality of service (QoS) will be negatively impacted if the decision system fails to choose the correct VM [10]. At present, the migration of VMs is determined by several factors, including the VM's resource utilization, the requirements of the target system, QoS and the server's use of its resources (central processing unit (CPU) and bandwidth). These approaches consider the load balance issue as a job-assignment optimization problem and concentrate primarily on creating optimization algorithms for quick merging. But they fail to consider for migration costs or the load on VMHs after balancing, they asume that the VM/VMH load is static, which limits their usefulness in real-world environments. Load balancing is a method of evenly distributing network traffic among a group of resources that support an application. Modern applications must process millions of users simultaneously and deliver the right text, videos, images, and other data to each user in a fast and reliable manner. It distributes traffic and workloads, ensuring that no single server or machine is underloaded, overloaded, or idle. Load balancing optimizes various constraint parameters such as execution time, response time and system stability to improve overall cloud performance.

Hence this is an efficient load balance using virtual machine migration hybrid optimization technique (HOT) in cloud computing is explained. The rest of the data is organized as follows: in section 2, the literature survey is explained; section 3 explains procedure of an efficient load balance using virtual machine migration HOT. The section 4 explains result analysis of this method. In section 5, the methodology is finally concluded.

2. LITERATURE SURVEY

Souravlas *et al.* [11] in order to increase the system's makespan and average response time in a cloud environment, suggest a task load balancing technique. The expected to utilizations for the VM, which are essential to the work allocation strategy is calculated from the balance state probabilities. The suggested task allocation technique is used by the load balancer (LBer), which operates as a central server in this architecture, to distribute incoming tasks across VMs in a fair and balanced way, then taking into consideration both their processing capabilities and their present state. According to results of this experiments, in terms of makespan and average reaction time, the proposed system performs better than present techniques, and resource usage while providing a lower level of imbalance.

Lahande *et al.* [12] focuses on the LBer mechanism using WorkflowSim and the Sipht task dataset. This first come first serve (FCFS), maximum–minimum (Max–Min), minimum completion time (MCT), minimum–minimum (Min–Min), and round-robin (RR) were the algorithms that the researchers used to balance the computing load of VMs. The experiment was carried out in four stages, with sixteen scenarios of varying task length at each stage. The results showed that the algorithms balanced the load 51.98%, 41.71%, 51.98%, 59.43%, and 52.17%, respectively. According to the study, to achieve optimal cloud resource utilization while maintaining the highest quality of service, LB should be enhanced with an intelligence mechanism. The research gap observed in this research is migration cost.

Dong *et al.* [13] suggests the host-enabled eBPF-based load balancing scheme (HEELS) organization concept for cloud data centers that combine edge computing. The host selection problem of task deployment is solved by applying the glowworm swarm optimization (GSO) algorithm and the examination of the task clustering approach. The strategy filters out large-resource tasks, utilizes the task for offloading technology, and utilizes the edge computing center's optimized GSO algorithm. The optimization of step size, combined with software communications architecture (SCA), improves global search ability and local convergence ability. Comparing HEELS to previous studies, the experimental results demonstrate improved load balancing and increased efficiency and green of the provided data center.

Kruekaew and Kimpan [14] offered the multi-objective task scheduling optimization based on the artificial bee colony algorithm (MOABCQ) method, which uses the Q-learning algorithm, a reinforcement learning methodology, and the artificial bee colony algorithm (ABC) to improve multi-objective task scheduling. In order to improve virtual machine throughput, optimize scheduling and resource use, develop load balancing among VMs based on makespan, cost, and resource utilization, the approach to cloud computing environments is a multi-objective method of work scheduling. Using CloudSim, the approach was compared to other load balancing and scheduling methods on three datasets: workload types include synthetic, random, and google cloud jobs (GoCJ). Then it cames to decreasing makespan, costs, imbalance, throughput, and average resource consumption, the MOABCQ method performed better than the other algorithms.

Lee *et al.* [15] created a high-performance load balancer that is simple to use with Kubernetes for Linux kernel traffic distribution, utilizing (eBPF eXpress Data Path (XDP)) eBPF/XDP. It evaluated utilizing Internet Mix/IMIX traffic streams and compared its performance with iptables dynamic network address translation (DNAT) and loopback, depending on the request for comment (RFC)2544 performance standard. The suggested load balancer performed better than iptables DNAT, with a small difference between the two, based on the results. The research gap observed in this research is migration cost.

Domanal *et al.* [16] in cloud computing environments, the hybrid bio-inspired algorithm provides efficient work scheduling and resource management. It provides tasks to VMs using modified particle swarm optimization, and then allocates resources using the recommended hybrid bio-inspired algorithm (modified particle swarm optimization (PSO)+modified cat swarm optimization (CSO)). According to peer research, the suggested hybrid algorithm has better efficiency, reliability, and lower average reaction times than benchmark algorithms based on experimental data and effective cloud resource utilization.

Garcia and Nafarrate [17] suggests using VM live migration to provide distributed problem-solving strategies for load management in data centers. To balance and reduce heterogeneous load costs, collaborative agents are provided with an energy-aware combination protocol and a load balancing protocol. They are given policies for migration, selection, host selection, and turning off/on hosts. The highest resource consumption imbalance is transferred from overloaded sites to underutilized hosts utilizing a novel load balancing strategy that is proposed. These methods are successful and efficient, according to empirical results. The research gap observed in this research is migration cost.

Zhang *et al.* [18] when the suggested method is used instead of other approaches like round-robin, min-min, and differential evolution, cloud service provider costs are effectively reduced and user task makespan is decreased. This improved differential evolution (IDE) approach addresses the challenge of VM allocation in cloud computing systems. The approach works better than these approaches, resulting in high user and provider satisfaction. The technique may be easily tested with CloudSim, this is a cloud simulation tool, and is applicable to industrial cloud computing systems.

Addya *et al.* [19] in a cloud federation, a framework for safe live VM movement is suggested. Its associated costs are examined for parallel, enhanced techniques. A CloudSim simulator is used to model the suggested architecture. Communication overhead, migration time, and downtime are the metrics that are taken into consideration for assessment. Additionally, a comparison and calculation of power usage are made for each strategy. This has been determined that enhanced techniques have the least amount of migration time and the least amount of downtime. Improved serial uses the least amount of power whereas parallel uses the most. The research gap observed in this research is accuracy.

Xu *et al.* [20] presents iAware, a minimal interference-aware virtual machine live migration technique. Through realistic workload experiments on a Xen virtualized cluster platform, it empirically captures the critical correlations between VM performance interference and important aspects that are practically accessible. iAware creates a basic multi-resource demand-supply model, which it uses to jointly estimate and minimize co-location, migration interference among VMs. To verify the performance gain and runtime overhead of iAware in comparison to the conventional interference-unaware virtual machine migration techniques, extensive experiments and large-scale simulations are carried out. Input/output, network, CPU, and scalability throughput are all measured. Furthermore, they show the ability of iAware is to work in balance with current VM scheduling or consolidation strategies in order to maintain performance while still achieving load balancing or power savings.

Shen and Chen [21] suggest a proactive load balancing approach for cloud datacenter virtual resource management that is based on the Markov decision process (MDP). Through supporting a VM in deciding the best way to transition to a lightly loaded state, the technique reduces load balancing overhead and service level agreement (SLA) violations. The authors also suggest improvement techniques, like a new MDP model that takes into consideration both VM migrations and a cloud profit-oriented reward system. Considering load balancing efficiency, long-term maintenance, and SLA violations, the algorithm outperforms existing reactive and proactive load balancing techniques, according to the results. They also highlight the efficiency of the suggested enhancement techniques.

Mandal *et al.* [22] a method for performing numerical tests with a lot of migration requests in multiple cloud environments, as well as the proper migration bandwidth and quantity of pre-copies. The results show that when compared to bandwidth provisioning strategies using maximum and minimum values, this method uses orders of magnitude less bandwidth and network resources than the maximum bandwidth option. In comparison to the minimum-bandwidth technique, it also achieves a significantly reduced migration duration.

Pradhan *et al.* [23] provided that the deep reinforcement learning with parallel particle swarm optimization (DRLPPSO), an efficient scheduling technique, to solve the load balancing problem and its many parameters quickly, accurately. In comparison to the modified PSO (MPSO), asynchronous advantage actor-critic (A3C), and deep Q-network (DQN) techniques, these experimental results demonstrate that the proposed scheduling algorithm has improves the reward by 17.5%, 12.6%, and 15.3% when the task set is 4,000. The research gap observed in this research is accuracy.

Laalaoui and Al-Omari [24] to get over the algorithm's space limitation, two methods are suggested: the direct move heuristic (DMH) and the iterative direct move heuristic (IDMH). Furthermore, they recommend two experimental investigations that were carried out using randomly generated instances of problems. The first experimental research examines cases of minor issues. It evaluates the effectiveness of the suggested algorithms and attempts to demonstrate the applicability of the modeling that is provided. Large-scale instances of problems are the focus of the second experimental research. It evaluates the IDMH heuristic's scalability performance. On problem instances with up to 800 VMs, the collected results demonstrate a good scalability performance.

Duan and Yang [25] provide a virtualization framework that takes into consideration these two issues at the same time. This architecture benefits from the growing use of OpenFlow protocols and distributed virtual switches (DVS). Firstly, virtual private clouds (VPCs) can have arbitrary traffic matrices supported by the architecture, which accommodates varied network communication patterns. The network interface bandwidth of a server is the only limitation on network flows. Secondly, the framework uses a complex connection construction method to provide load balancing. They focus on this framework on the fat-tree design, which is one of the most common types of data center architecture is available on today.

3. AN EFFICIENT LOAD BALANCE USING VIRTUAL MACHINE MIGRATION HOT IN CLOUD COMPUTING

In this section, block diagram of an efficient load balance using VM migration hybrid optimization technique in cloud computing is observed in Figure 1. Implementing the VM requests in step with the physical machines (PMs) resource availability on these hosts is known as VM placement. VMs need to be distributed effectively so that no system or request is left waiting for a response from the cloud. Making the best use of the resources at hand is the main objective of the VM placement task. The load and scheduling receive the VM request. On the cloud, VMs are scheduled to optimize their utilization. In order to maximize advantages for cloud service providers, this scheduling helps in improving service quality. Through utilizing cloud computing services and saving energy, companies can decrease costs. To ensure that no one server or computing is under-loaded, overloaded, or idle, load balancing is used in cloud computing to share workloads and traffic. Load balancing enhances overall cloud performance by optimizing a number of limited characteristics, including response time, execution speed, and stability of the system. The VM manager is provided that schedule and load. A collection of tools called VM could be used to control the operating systems of large fleets of VMs running Linux and Windows on Compute Engine. By automating tasks, VM manager improves productivity and reduces the operational load of managing these VMs. As the VM manger carries towards virtual machine load model. Predicting VM load is an essential task in cloud computing.

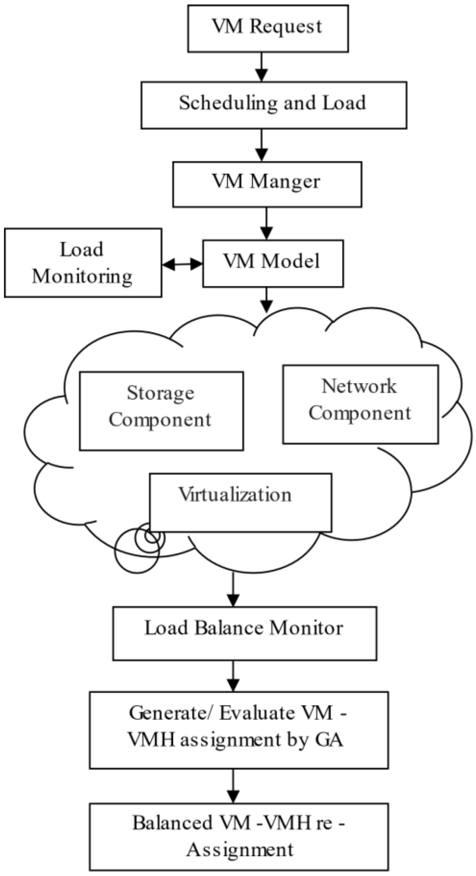

Figure 1. Block diagram of an efficient load balance using virtual machine migration HOT in cloud computing

Accurate prediction of virtual machine load can enhance resource allocation, reduce costs, and improve service quality. Then VM load model will monitor the load. Then it is stored in cloud, that the data is given to storage component, network component a virtualization. The back-end storage component houses the data needed to run programs. Large volumes of data can be stored and managed on the cloud with the help of flexible, scalable storage services offered by a majority of cloud service providers, yet exact options could differ. Establishing computer networks with both hardware and software components are requires the use of network components. Network components that are often used hubs, switches, cables, routers, networking operating systems, and other components. The ability to construct virtual versions of servers, storage, networks, and other physical machines is known as virtualization technology. On a single physical computer, virtual software runs several VMs concurrently by simulating the operations performed by physical hardware. Once more, data will be monitored by load balance monitor. Genetic algorithm (GA) will be assigned for the VM-VMH model to generate the result. GA are used in load balancing strategies. While attempting to minimize the duration of a given task set, the algorithm at balancing the load on the cloud infrastructure. If any errors then again it is re-assigned for balanced VM-VMH.

4. RESULT AND OBSERVATIONS

In this section, performance analysis of efficient load balance using virtual machine migration HOT in cloud computing is observed in Table 1. In Table 1, the comparison between MOABCQ and hybrid optimization is observed in terms of accuracy, energy consumption and migration cost parameters. In Figure 2, accuracy comparison graph between hybrid optimization and MOABCQ. In this graph X-axis demonstrates VM migration and Y-axis demonstrates accuracy. The accuracy of hybrid optimization is higher when compared with MOABCQ.

Table 1. Performance analysis

Parameters	MOABCQ	Hybrid optimization
Accuracy	96.7	98.2
Energy consumption	89.6	82.7
Migration cost	94597	89756

In this Figure 3, the graph X-axis demonstrates virtual machine migration and Y-axis demonstrates energy consumption. Figure 3, shows the energy consumption graph between hybrid optimization and MOABCQ in efficient load balance using virtual machine migration HOT in cloud computing. MOABCQ uses a lot of energy when compared with hybrid optimization.

The migration cost is reduced in hybrid optimization in efficient load balance using VM migration HOT in cloud computing. In this graph X-axis demonstrates VM migration and Y-axis demonstrates migration cost. Between hybrid optimization and MOABCQ algorithm there is a graphical representation of the migration cost shown in Figure 4.

VM migration process is important because it moves VMs from one physical server to another to balance load and save power. Therefore, by using this hybrid optimization is load balanced accurately. Even accuracy is increased, energy consumption and migration cost is reduced when compared with other methods.

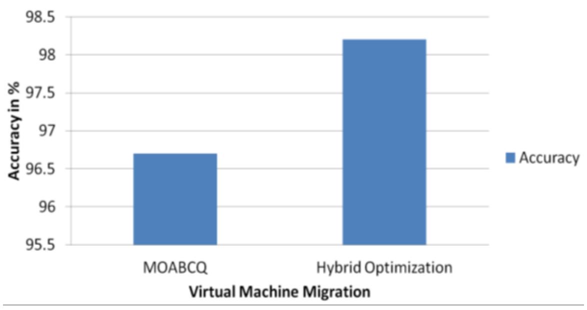

Figure 2. Accuracy comparison graph

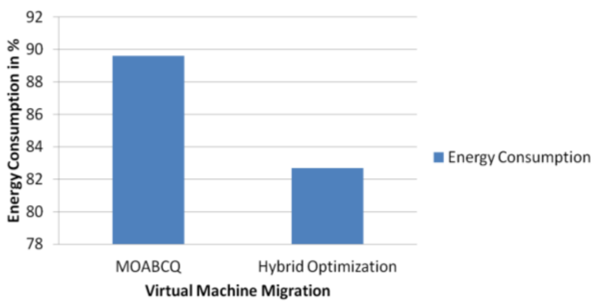

Figure 3. Energy consumption graph

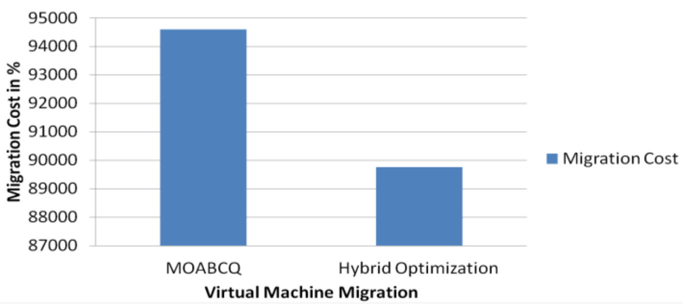

Figure 4. Migration cost comparison graph

5. CONCLUSION

In this section, an efficient load balance using VM migration HOT in cloud computing is observed. Using gene expression programming (GEP), symbolic regression models of VMs are developed using measured VM loads and the corresponding resource parameters. By using GEP models to estimate VMH loads and recommending VM migration for load balancing, GA determines the best possible combination of VM-VMH assignment. Traditional computing system mostly uses centralized server system but cloud system uses distributed and grid computing systems to manage resources with effective time and energy. Management of resources in effective time and energy saving mechanism both are not parallel easy to manage. Algorithm balances the load among VMs and achieve on time migration. Load migration should be proper scheduled, synchronized and also able to simultaneously load distribution. Therefore, by using this hybrid optimization to load balanced accurately. Even accuracy is increased, energy consumption and migration cost are reduced when compared with other methods. By using this method additional server requirements are avoided when the load is high and also the cloud resources can be used properly. Hence, this model achieves better results in terms of accuracy, energy consumption and migration cost. In future, this algorithm could be further extended in future by considering other elements such as accessibility, security, and scalability.

REFERENCES

[1] L.-H. Hung, C.-H. Wu, C.-H. Tsai, and H.-C. Huang, "Migration-based load balance of virtual machine servers in cloud computing by load prediction using genetic-based methods," *IEEE Access*, vol. 9, pp. 49760–49773, 2021, doi: 10.1109/ACCESS.2021.3065170.

[2] H. Shen and L. Chen, "A resource usage intensity aware load balancing method for virtual machine migration in cloud datacenters," *IEEE Transactions on Cloud Computing*, vol. 8, no. 1, pp. 17–31, Jan. 2020, doi: 10.1109/TCC.2017.2737628.

[3] M. A. Shahid, N. Islam, M. M. Alam, M. M. Su'ud, and S. Musa, "A comprehensive study of load balancing approaches in the cloud computing environment and a novel fault tolerance approach," *IEEE Access*, vol. 8, pp. 130500–130526, 2020, doi: 10.1109/ACCESS.2020.3009184.

[4] S. K. Maurya, S. Malik, and N. Kumar, "Virtual machine tree task scheduling for load balancing in cloud computing," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 30, no. 1, pp. 388–393, Apr. 2023, doi: 10.11591/ijeecs.v30.i1.pp388-393.

[5] K. Sekaran, M. S. Khan, R. Patan, A. H. Gandomi, P. V. Krishna, and S. Kallam, "Improving the response time of m-learning and cloud computing environments using a dominant firefly approach," *IEEE Access*, vol. 7, pp. 30203–30212, 2019, doi: 10.1109/ACCESS.2019.2896253.

[6] S. Sotiriadis, N. Bessis, C. Amza, and R. Buyya, "Elastic load balancing for dynamic virtual machine reconfiguration based on vertical and horizontal scaling," *IEEE Transactions on Services Computing*, vol. 12, no. 2, pp. 319–334, Mar. 2019, doi: 10.1109/TSC.2016.2634024.

[7] M. Liaqat, A. Naveed, R. L. Ali, J. Shuja, and K.-M. Ko, "Characterizing dynamic load balancing in cloud environments using virtual machine deployment models," *IEEE Access*, vol. 7, pp. 145767–145776, 2019, doi: 10.1109/ACCESS.2019.2945499.

[8] M. Junaid, A. Sohail, A. Ahmed, A. Baz, I. A. Khan, and H. Alhakami, "A hybrid model for load balancing in cloud using file type formatting," *IEEE Access*, vol. 8, pp. 118135–118155, 2020, doi: 10.1109/ACCESS.2020.3003825.

[9] J. Zhao, K. Yang, X. Wei, Y. Ding, L. Hu, and G. Xu, "A heuristic clustering-based task deployment approach for load balancing using bayes theorem in cloud environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 305–316, Feb. 2016, doi: 10.1109/TPDS.2015.2402655.

[10] M. Junaid *et al.*, "Modeling an optimized approach for load balancing in cloud," *IEEE Access*, vol. 8, pp. 173208–173226, 2020, doi: 10.1109/ACCESS.2020.3024113.

[11] S. Souravlas, S. D. Anastasiadou, N. Tantalaki, and S. Katsavounis, "A fair, dynamic load balanced task distribution strategy for heterogeneous cloud platforms based on markov process modeling," *IEEE Access*, vol. 10, pp. 26149–26162, 2022, doi: 10.1109/ACCESS.2022.3157435.

[12] P. V. Lahande, P. R. Kaveri, J. R. Saini, K. Kotecha, and S. Alfarhood, "Reinforcement learning approach for optimizing cloud resource utilization with load balancing," *IEEE Access*, vol. 11, pp. 127567–127577, 2023, doi: 10.1109/ACCESS.2023.3329557.

[13] Y. Dong, G. Xu, Y. Ding, X. Meng, and J. Zhao, "A 'Joint-Me' task deployment strategy for load balancing in edge computing," *IEEE Access*, vol. 7, pp. 99658–99669, 2019, doi: 10.1109/ACCESS.2019.2928582.

[14] B. Kruekaew and W. Kimpan, "Multi-objective task scheduling optimization for load balancing in cloud computing environment using hybrid artificial bee colony algorithm with reinforcement learning," *IEEE Access*, vol. 10, pp. 17803–17818, 2022, doi: 10.1109/ACCESS.2022.3149955.

[15] J.-B. Lee, T.-H. Yoo, E.-H. Lee, B.-H. Hwang, S.-W. Ahn, and C.-H. Cho, "High-performance software load balancer for cloud-native architecture," *IEEE Access*, vol. 9, pp. 123704–123716, 2021, doi: 10.1109/ACCESS.2021.3108801.

[16] S. G. Domanal, R. M. R. Guddeti, and R. Buyya, "A hybrid bio-inspired algorithm for scheduling and resource management in cloud environment," *IEEE Transactions on Services Computing*, vol. 13, no. 1, pp. 3–15, Jan. 2020, doi: 10.1109/TSC.2017.2679738.

[17] J. O. G.- Garcia and A. R.- Nafarrate, "Collaborative agents for distributed load management in cloud data centers using live migration of virtual machines," *IEEE Transactions on Services Computing*, vol. 8, no. 6, pp. 916–929, Nov. 2015, doi: 10.1109/TSC.2015.2491280.

[18] P. Zhang, M. Zhou, and X. Wang, "An intelligent optimization method for optimal virtual machine allocation in cloud data centers," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 4, pp. 1725–1735, Oct. 2020, doi: 10.1109/TASE.2020.2975225.

[19] S. K. Addya, A. K. Turuk, A. Satpathy, B. Sahoo, and M. Sarkar, "A strategy for live migration of virtual machines in a cloud federation," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2877–2887, Sep. 2019, doi: 10.1109/JSYST.2018.2872580.

[20] F. Xu, F. Liu, L. Liu, H. Jin, B. Li, and B. Li, "iAware: making live migration of virtual machines interference-aware in the cloud," *IEEE Transactions on Computers*, vol. 63, no. 12, pp. 3012–3025, Dec. 2014, doi: 10.1109/TC.2013.185.

[21] H. Shen and L. Chen, "Distributed autonomous virtual resource management in datacenters using finite-markov decision process," *IEEE/ACM Transactions on Networking*, vol. 25, no. 6, pp. 3836–3849, Dec. 2017, doi: 10.1109/TNET.2017.2759276.

[22] U. Mandal, P. Chowdhury, M. Tornatore, C. U. Martel, and B. Mukherjee, "Bandwidth provisioning for virtual machine migration in cloud: strategy and application," *IEEE Transactions on Cloud Computing*, vol. 6, no. 4, pp. 967–976, Oct. 2018, doi: 10.1109/TCC.2016.2545673.

[23] A. Pradhan, S. K. Bisoy, S. Kautish, M. B. Jasser, and A. W. Mohamed, "Intelligent decision-making of load balancing using deep reinforcement learning and parallel PSO in cloud environment," *IEEE Access*, vol. 10, pp. 76939–76952, 2022, doi: 10.1109/ACCESS.2022.3192628.

[24] Y. Laalaoui and J. Al-Omari, "A planning approach for reassigning virtual machines in IaaS clouds," *IEEE Transactions on Cloud Computing*, vol. 8, no. 3, pp. 685–697, 2018, doi: 10.1109/TCC.2018.2826548.

[25] J. Duan and Y. Yang, "A load balancing and multi-tenancy oriented data center virtualization framework," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 8, pp. 2131–2144, 2017, doi: 10.1109/TPDS.2017.2657633.

BIOGRAPHIES OF AUTHORS

Saravanan Madderi Sivalingam is a professor at the Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, India. He is an accomplished researcher and academic in the field of computer science and engineering. He has authored 2 books, edited 4 papers, published 153 papers in refereed international journals presented 26 papers in refereed conference proceedings and 16 patents published. He also gave 21 major invited contributions and/or technical reports abstracts and/or papers read. Having Cloud Foundation Certificate and Data Analytics IBM Cognos Certificate. Since 2017 he has served at Haramaya University, East Africa as a professor in the School of Computing for two years. He is a member of IEEE, ISTE, and IET as well as a Student Branch Councilor of IEEE and Innovation Ambassador of the Institution's Innovation Council (IIC), SIMATS. He expertise lies in artificial intelligence, data science, and cloud-based technologies. He has been recognized for his groundbreaking work in developing a less-cost cabinet dyeing process using process mining techniques for this developed a new algorithm called "LinkRuleMiner". He leads a dynamic research group focused on advancing artificial Intelligence-based products. His lab's innovative research has been published in leading peer-reviewed journals. He also having the best faculty and researcher awards from national and international societies. He actively mentors graduate students and collaborates with industry partners to bridge the gap between academia and practical applications. He can be contacted at email: saranenadu@gmail.com.

Pavan Kumar Prathapagiri received the B.Tech. degree in information technology from St. Ann's College of Engineering and Technology, JNTUK University, Kakinada, Andhra Pradesh in 2010 and M.Tech. Degree in computer science and engineering from Vignan University, AP in 2013, he is an assistant professor at the Department of Computer Science and Engineering, Guru Nanak Institutions Technical Campus, India. He has published 1 papers in refereed conference proceedings published. His research includes load balancing in cloud computing. He can be contacted at email: pavanathraya@gmail.com.